BIOINFORMATICS ORIGINAL PAPER
Vol. 25 no. 7 2009, pages 910–917
doi:10.1093/bioinformatics/btp066

Systems biology

Simulation of large-scale rule-based models

Joshua Colvin[1], Michael I. Monine[2], James R. Faeder[3], William S. Hlavacek[2,4], Daniel D. Von Hoff[5] and Richard G. Posner[1,6,*]

[1]Computational Biology Division, Translational Genomics Research Institute, Phoenix, AZ 85004, [2]Theoretical Division and Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM 87545, [3]Department of Computational Biology, University of Pittsburgh School of Medicine, Pittsburgh, PA 15260, [4]Department of Biology, University of New Mexico, Albuquerque, NM 87131, [5]Clinical Translational Research Division, Translational Genomics Research Institute, Phoenix, AZ 85004 and [6]Department of Biological Sciences, Northern Arizona University, Flagstaff, AZ 86011, USA

Received on September 26, 2008; revised on January 13, 2009; accepted on January 27, 2009

Advance Access publication February 11, 2009

Associate Editor: Olga Troyanskaya

ABSTRACT

Motivation: Interactions of molecules, such as signaling proteins, with multiple binding sites and/or multiple sites of post-translational covalent modification can be modeled using reaction rules. Rules comprehensively, but implicitly, define the individual chemical species and reactions that molecular interactions can potentially generate. Although rules can be automatically processed to define a biochemical reaction network, the network implied by a set of rules is often too large to generate completely or to simulate using conventional procedures. To address this problem, we present DYNSTOC, a general-purpose tool for simulating rule-based models.

Results: DYNSTOC implements a null-event algorithm for simulating chemical reactions in a homogenous reaction compartment. The simulation method does not require that a reaction network be specified explicitly in advance, but rather takes advantage of the availability of the reaction rules in a rule-based specification of a network to determine if a randomly selected set of molecular components participates in a reaction during a time step. DYNSTOC reads reaction rules written in the BioNetGen language which is useful for modeling protein–protein interactions involved in signal transduction. The method of DYNSTOC is closely related to that of StochSim. DYNSTOC differs from StochSim by allowing for model specification in terms of BNGL, which extends the range of protein complexes that can be considered in a model. DYNSTOC enables the simulation of rule-based models that cannot be simulated by conventional methods. We demonstrate the ability of DYNSTOC to simulate models accounting for multisite phosphorylation and multivalent binding processes that are characterized by large numbers of reactions.

Availability: DYNSTOC is free for non-commercial use. The C source code, supporting documentation and example input files are available at http://public.tgen.org/dynstoc/.

Contact: dynstoc@tgen.org

Supplementary information: Supplementary data are available at *Bioinformatics* online.

*To whom correspondence should be addressed.

1 INTRODUCTION

Models for biochemical reaction networks are commonly specified in terms of (i) a list of reactions; (ii) a visual layout of the reactions in a system or a reaction scheme diagram, which may be enhanced using annotative iconography (Kitano *et al.*, 2005); or (iii) a set of ordinary differential equations (ODEs), in which one equation is included for each possible chemical species in a network. Such models can also be specified in terms of reaction rules (Hlavacek *et al.*, 2006), which provide a high-level representation of the molecular interactions in a system that give rise to individual reactions and chemical species. Rules are particularly useful when one wishes to account for molecular substructure and/or site-specific details of molecular interactions in a model, as illustrated by the didactic example of Danos (2007). The BioNetGen language (BNGL) (Blinov *et al.*, 2006; Faeder *et al.*, 2005b, in press) and the closely related κ-calculus (Danos and Laneve, 2004; Danos *et al.*, 2007a) are examples of formal languages that have been developed for precisely specifying models for biochemical systems in terms of reaction rules. Other related modeling frameworks include dynamical grammars (Mjolsness and Yosiphon, 2006), ρ_{bio}-calculus (Andrei and Kirchner, 2008), and BlenX (Dematté *et al.*, 2008). BNGL can be processed by the BioNetGen software package to generate a reaction network automatically from a set of BNGL-encoded rules (Blinov *et al.*, 2004; Faeder *et al.*, 2005a; http://bionetgen.org/). The rule-derived network can then be simulated using conventional methods that take a reaction network as input. Rules have been used most commonly to model signal-transduction systems (Hlavacek *et al.*, 2006), but rules have been used to model other types of biochemical systems as well (for example, see Mu *et al.*, 2007).

A set of rules often implies a vast reaction network (Danos *et al.*, 2007a; Hlavacek *et al.*, 2003, 2006). Deriving a network from such a set of rules is computationally expensive, in large part because of the need to routinely solve graph isomorphism problems (Blinov *et al.*, 2006). Moreover, simulating a model for such a network (if it can be fully derived) may be impractical with conventional methods. For example, the computational cost of numerically integrating ODEs, which is often the most efficient approach for simulating

a small reaction network, depends non-linearly on N, the number of ODEs (i.e. the number of chemical species in a network). For stiff ODEs, the cost of numerical integration typically scales with N^3, and simulating a system for which N is larger than 10^4–10^5 can be prohibitively expensive. The computational cost of stochastic simulation methods, such as Gillespie's method and most subsequent improvements of this method (Gillespie, 2007; Li *et al*., 2008), depends on M, the number of reactions in a system. Schulze (2007) and Slepoy *et al*. (2008) have recently described stochastic simulation methods with costs independent of M, but to apply either of these methods to simulate a rule-based model, one must first generate a network from rules, which again is costly.

The expense of network generation can be avoided or reduced in several ways. It can be reduced by terminating network generation before the full list of potential reactions is obtained and then simulating the resulting partial network, but the accuracy of results is uncertain. Network generation is minimized in a principled way in the on-the-fly method of stochastic simulation (Faeder *et al*., 2005a; Gillespie, 2007; Lok and Brent, 2005), in which only reactions that connect populated and reachable chemical species are generated. However, on-the-fly simulation of a highly branched reaction network is significantly slowed by the costs of network generation (Hlavacek *et al*., 2006; Yang *et al*., 2008). Simulation is sometimes made tractable by methods for reducing the size of rule-based models (Borisov *et al*., 2005; Conzelmann *et al*., 2006). However, model reduction methods are not generally applicable. New approaches and software tools are needed for simulation of large-scale rule-based models, i.e. models of systems described by rules that imply large numbers of possible chemical species and reactions.

To address this need, two closely related particle- or agent-based methods have recently been developed for simulating rule-based models (Danos *et al*., 2007b; Yang *et al*., 2008). In these methods, a time increment is sampled from an exponential distribution, a rule is selected from among a weighted list of rules, just as reactions are sampled in Gillespie's method, and the selected rule is used to generate a reaction event (i.e. to select reactants to participate in a type of reaction defined by the selected rule). Reaction network generation is avoided. Software implementing the method of Danos *et al*. (2007b), called Kappa Factory, is available from Plectix BioSystems, Inc. (W.Fontana, personal communication). This software is capable of processing model specifications defined using κ. Similar general-purpose software implementing the method of Yang *et al*. (2008), which extends the method of Danos *et al*. (2007b), will be available soon for simulating BNGL-encoded models.

Here, to further address the need for tools that can simulate large-scale rule-based models, we present DYNSTOC, an open-source software that implements an extension of the StochSim simulation method (Le Novère and Shimizu, 2001; Morton-Firth and Bray, 1998; Shimizu and Bray, 2001). The method implemented in StochSim is an agent-based null-event simulation procedure. In the extension of this procedure, BNGL-encoded reaction rules play an integral role, and there is no requirement for an a priori specification of the individual reactions implied by the rules. The main difference between DYNSTOC and StochSim is in the ability of DYNSTOC to use a set of rules specified in an expressive model-specification language to assess whether randomly selected molecular components are able to participate in a reaction. This difference greatly expands the range of rule-based models that can be simulated using the StochSim/DYNSTOC approach.

2 METHOD

The conventions of the model-specification language BNGL, the graphical foundations of BNGL, the graph-rewriting approach used to interpret BNGL-encoded reaction rules and the StochSim simulation approach have been described in detail elsewhere (Blinov *et al*., 2006; Faeder *et al*., 2005b, in press; Hlavacek *et al*., 2006; Morton-Firth and Bray, 1998; Shimizu and Bray, 2001). Below, we present a generalization of the StochSim simulation procedure that incorporates the representational conventions of BNGL. The StochSim method, a null-event kinetic Monte Carlo (KMC) method, has sometimes been characterized as an approximate method; it is not. Properly parameterized, it produces the same statistical distribution of events as a rejection-free KMC method, such as Gillespie's method (for an informative review of null-event and rejection-free KMC methods; see Chatterjee and Vlachos, 2007).

2.1 Representational conventions

Molecules and molecular complexes are represented by graphs, and molecular interactions or reaction types are represented by graph-rewriting rules. Graphs are comprised of nodes, labels associated with nodes and edges that connect nodes. Nodes represent the reactive molecular components of a system (e.g. sites and domains of proteins), which are tracked individually during a simulation. In other words, each node is associated with a unique index. Components may be associated with multiple internal states (e.g. a tyrosine residue may be labeled as either phosphorylated or unphosphorylated). Labels give the names of components and their internal states. Edges represent bonds between components. A graph-rewriting rule identifies the necessary and sufficient properties of reactants in a particular type of reaction, the products that result from this type of reaction given a set of reactants, and a rate law for reactions defined by the rule. In the simulation procedure presented here, the rate law associated with a rule is used to determine the probability that a reaction occurs within a given fixed time step. We assume that rate laws associated with rules characterize elementary reactions. In short, we use standard BNGL to represent molecules and molecular interactions with the added feature of unique node indices. These indices are used only for tracking purposes in the simulation procedure described below, and the tracking index associated with a node does not affect its reactivity in any way.

2.2 Algorithm

The simulation procedure is comprised of the following steps, which are repeated until a specified stopping criterion is satisfied.

(1) The current time is incremented by a fixed time step Δt, during which at most one reaction is allowed to occur. Selection of Δt is discussed below in Section 2.5. The value of Δt determines the resolution with which one can determine when events occur. With each time increment, a reaction is attempted as follows.

(2) A decision is made to select one or two nodes from among the nodes representing reactive components in the system of interest. [We focus on reactions that affect the connection(s) and/or internal state(s) of one or two nodes. It is straightforward to extend the method as presented here to enable consideration of additional types of reactions, such as termolecular reactions and synthesis, degradation and transport reactions.] One node is selected with probability π_1, and two nodes are selected with probability $\pi_2 = 1 - \pi_1$. Selection of π_1 is discussed below in Section 2.4.

(3) Depending on the outcome of the previous step, either a single node or a set of two nodes is selected randomly. Each node is selected with uniform probability. The selection procedure is such that the same node can be selected twice, but if a node is selected twice, the following steps are skipped and a null event is said to occur during the time step.

(4) The selected node or pair of nodes is checked against all reaction rules specified for the reaction system of interest to determine if the node or pair of nodes qualifies as a reactant or set of co-reactants in one or more types of reactions defined by the rules. This procedure often requires that the local environment of a selected node be compared against information included in a rule, as described elsewhere (Blinov *et al*., 2006), to determine if the environment is permissive for reaction. For example, if phosphorylation of a tyrosine depends on co-localization with a kinase, the presence or absence of the kinase in the local environment of the tyrosine must be established before one can determine whether the tyrosine can be phosphorylated. At the end of this step, a list of possible reactions has been identified, including potential reactants. If no reaction is possible, the following steps are skipped and a null event occurs.

(5) Each possible reaction is labeled with an integer index $r \in [1,\ldots,M_j]$, where M_j is the number of possible reactions for iteration j of the simulation procedure, and a probability p_r is calculated for each reaction. The calculation of reaction probabilities is discussed below in Section 2.3.

(6) A uniform deviate $\rho \in (0,1)$ is generated and used to determine which, if any, of the M_j possible reactions occurs by finding the smallest integer $R \leq M_j$ that satisfies $\sum_{r=1}^{R} p_r \geq \rho$, where p_r is the probability calculated in Step 5 for reaction r. If no value of R satisfies the inequality, the following step is skipped and a null event occurs.

(7) The graph-rewriting operation of the rule defining the reaction with index R is applied to the graph(s) representing the reactant(s) identified in Step 4. The graph-rewriting process has been described elsewhere (Blinov *et al*., 2006).

2.3 Calculation of the probability of accepting a reaction

If a rule indicates that a set of nodes selected in Step 3 of the simulation procedure (see Section 2.2) can participate in a reaction r (e.g. a rule indicates that a selected pair of nodes can be connected by an edge or a rule indicates that a selected single node can change its internal state), the reaction is accepted with probability p_r (Step 6). In general, the value of p_r is chosen such that $p_r P_r/\Delta t = v_r N_A V$, where v_r is the rate law (inherited from the governing rule) that gives the molar rate of reaction r (in units such as Ms^{-1}), N_A is Avogadro's number, V is the volume of the reaction system and P_r is the probability of selecting a set of nodes that can participate in reaction r in Step 3 of the simulation procedure. In other words, p_r is chosen such that the expected number of reactants consumed in reaction r per time step as a result of applying the simulation procedure, which is given by $p_r P_r/\Delta t$, matches the corresponding physicochemical turnover rate, which is given by $v_r N_A V$.

Shimizu and Bray (2001) presented a derivation of expressions that can be used to determine the acceptance probabilities that should be used for two special types of reactions. Following the approach of Shimizu and Bray (2001), we can derive similar expressions for other types of reactions. Below, we give expressions used by DYNSTOC for four types of reactions: (I) state-change reactions, (II) bimolecular association reactions, (III) dissociation reactions and (IV) unimolecular association reactions (i.e. reactions in which two parts of the same molecule or molecular complex are connected). We will refer to these types of reactions as Types I–IV. Below, we will assume that these reactions are characterized by mass–action rate laws.

The probability of accepting a state-change reaction r, which we will denote as p_{I}^r, is given by

$$p_{\mathrm{I}}^r = \begin{cases} \frac{\kappa_1^r n \Delta t}{\pi_1} & \text{if} \quad \pi \leqslant \pi_1 \\ 0 & \text{if} \quad \pi > \pi_1 \end{cases} \tag{1}$$

where $\pi \in (0,1)$ is a uniform deviate generated in Step 2 of the simulation procedure, $\kappa_1^r = k_{\mathrm{I}}^r$, k_{I}^r is a rate constant (with units such as s^{-1}) that characterizes the rate at which a component changes state through reaction r, n is the total number of components (nodes) in the reaction system of interest and Δt is the fixed time step used in the simulation procedure. Recall that $\pi_1 = 1 - \pi_2$ is the probability of deciding to inspect a single node (versus a pair of nodes) in Step 2 of the simulation procedure. Thus, according to Equation (1), Type I reactions are fired only when the decision is made to select a single node in Step 2 of the simulation procedure.

The probability of accepting a bimolecular association reaction r, which we will denote as p_{II}^r, is given by

$$p_{\mathrm{II}}^r = \begin{cases} 0 & \text{if} \quad \pi \leqslant \pi_1 \\ \frac{\kappa_2^r n^2 \Delta t}{2\pi_2} & \text{if} \quad \pi > \pi_1 \end{cases} \tag{2}$$

where $\pi \in (0,1)$ is a uniform deviate, $\kappa_2^r = k_{\mathrm{II}}^r/(N_A V)$ and k_{II}^r is a rate constant (with units such as $\mathrm{M}^{-1}\ \mathrm{s}^{-1}$) that characterizes the rate at which two (distinct) components associate through reaction r. The factor of 2 in the denominator of the expression for p_{II}^r arises because there are two ways (ordered sequences) in which a pair of nodes representing reactive components can be selected in Step 3 of the simulation procedure.

The probability of accepting a dissociation reaction r, which we will denote as p_{III}^r, is given by the right-hand side of Equation (1), except with κ_1^r redefined as follows: $\kappa_1^r = k_{\mathrm{III}}^r/2$, where k_{III}^r is a rate constant (with units such as s^{-1}) that characterizes the rate at which two (distinct) components dissociate through reaction r. Note that a dissociation reaction can be uniquely identified in two ways: by selecting either of the two components that are bound to each other when the choice is made in Step 2 of the simulation procedure to inspect a single node ($\pi \leq \pi_1$) or by selecting two mutually bound components when the choice is made to select a pair of components in Step 2 of the simulation procedure ($\pi > \pi_1$). For the latter case, we arbitrarily set $p_{\mathrm{III}}^r = 0$ because it is more efficient to identify reactions through selection of a single node than a pair of nodes. For the former case, the factor of $1/2$ that multiplies k_{III}^r in the expression for κ_1^r above arises because there are two ways that a single node can be selected to uniquely identify a dissociation reaction.

The probability of accepting a unimolecular (intramolecular) association reaction r (e.g. a reaction that connects two ends of a polymer chain to form a ring), which we will denoted as p_{IV}^r, is given by the right-hand side of Equation (2), except with κ_2^r redefined as follows: $\kappa_2^r = k_{\mathrm{IV}}^r$, where k_{IV}^r is a rate constant (with units such as s^{-1}) that characterizes the rate at which two (distinct) components of the same molecule or molecular complex associate through reaction r. The factor of 2 in the denominator of the expression for p_{IV}^r arises because there are two ways (ordered sequences) in which a pair of nodes representing reactive components can be selected. We arbitrarily set $p_{\mathrm{IV}}^r = 0$ when only a single node is inspected in Step 2 of the simulation procedure, because in general, selection of a single node is insufficient to uniquely identify a reaction of this type.

The above expressions for acceptance probabilities are provided for purposes of illustration. One can easily derive additional expressions from the general relation $p_r P_r/\Delta t = v_r N_A V$ by following the approach of Shimizu and Bray (2001). DYNSTOC is capable of handling the most common reaction types that can be defined using BNGL—see the examples available at the DYNSTOC web site. An error message is produced if DYNSTOC encounters a reaction type that it does not recognize.

2.4 Selection of the number of nodes to inspect

In Step 2 of the simulation procedure (see Section 2.2), a decision is made to inspect either one or two nodes. This decision depends on the value chosen for

π_1 and it determines the types of reactions that are subsequently considered in the simulation procedure. We will refer to reactions that are considered after the selection of a single node as Type 1 reactions. Likewise, we will refer to reactions that are considered after the selection of a pair of nodes as Type 2 reactions. Let us assume that both Type 1 and Type 2 reactions are possible; otherwise, we can set $\pi_1 = 0$ or 1 trivially. The value of π_1 may be set arbitrarily, but for efficiency, the value should be chosen to minimize null events. Following the approach of Morton-Firth and Bray (1998), we adopt the strategy of setting algorithmic parameters so that $p_1^{\max} = p_2^{\max} = 1$ to minimize null events, where

$$p_i^{\max} = \frac{n^i \Delta t}{i\pi_i} \kappa_i^{\max} \tag{3}$$

for $i \in [1,2]$. In Equation (3), $p_1^{\max}(p_2^{\max})$ denotes the largest probability of accepting a Type 1 (2) reaction at any step during a simulation, and accordingly, $\kappa_i^{\max}$ denotes the maximal value of $\sum_{r=1}^{M_j} \kappa_i^r$ that can be calculated for a list of reactions identified in Step 4 of the simulation procedure for any iteration j of the procedure in which a decision is made in Step 2 to inspect $i \in [1,2]$ nodes. In other words, to minimize the occurrence of null events, we normalize the largest cumulative probability of selecting a reaction in Step 6, regardless of whether one or two nodes are inspected in Step 3.

By setting $p_1^{\max} = p_2^{\max}$ and then using Equation (3) and the relation $\pi_2 = 1 - \pi_1$, we find the following expression for the optimal value of π_1:

$$\pi_1 = \frac{\kappa_1^{\max}}{\kappa_1^{\max} + \kappa_2^{\max} n/2} \tag{4}$$

Note that this expression does not depend on the time step Δt. Also note that the values of $\kappa_1^{\max}$ and $\kappa_2^{\max}$ depend on the connectivity of the reaction network being simulated as well as the factors, such as rate constants, indicated in Section 2.3.

By introducing the concept of pseudo nodes to set the value of (Morton-Firth and Bray, 1998), we can ensure that only three random numbers are generated in the simulation procedure for each time step instead of three or four. In this approach, π_1 is related to the number of pseudo nodes, n_0, as follows: $\pi_1 = n_0/(n+n_0)$. A suboptimal value of π_1 is likely to result from this procedure because n_0 is necessarily an integer. In this case, the optimal time step (given below) must be reduced. For example, as can be easily confirmed, if π_1 is less than the optimal value given by Equation (4) by a factor of $1-\varepsilon$, where $0 < \varepsilon < 1$, then the optimal time step must be reduced by the same factor. Note that $1-\varepsilon$ is the relative error introduced by using an integer pseudo node count that gives a value for π_1 less than that given by Equation (4).

2.5 Selection of the time step

In Gillespie's method, the time step is sampled from an exponential distribution and a reaction occurs at each time step. In contrast, DYNSTOC (as well as StochSim) uses a fixed time step and rejection sampling. In other words, this approach introduces null events, i.e. time steps in which no reactions occur. The time step must be carefully chosen. If the time step is too large, more than one reaction is likely to occur (in the physical system) during a step and the accuracy of simulation is degraded. If the time step is too small, accuracy is ensured but at the expense of computational efficiency, because the cost of null events is wasted. These events do not change the state of a system.

When is given by Equation (4), we can use $p_1^{\max} = 1$ and Equation (3) to find the largest time step Δt that can be used in a simulation without introducing error:

$$\Delta t = \frac{1}{\kappa_1^{\max} n + \kappa_2^{\max} n^2/2} \tag{5}$$

It should be noted that, although in this equation does not explicitly depend on the choice of, Equation (5) is valid only if is given by Equation (4).

DYNSTOC uses Equations (4) and (5) to automatically set the values of π_1 and Δt on the basis of estimated values of $\kappa_1^{\max}$ and $\kappa_2^{\max}$, which are initially obtained from an inspection of the rules to be used in a simulation. The cumulative probability of accepting a reaction in Step 6 of the simulation procedure is checked at every iteration of the procedure and if this quantity is ever found to be greater than 1, DYNSTOC generates an error message reporting how Δt should be manually rescaled to normalize the cumulative probability.

3 DEMONSTRATION AND VALIDATION

To validate DYNSTOC, we simulated a number of rule-based models and compared the results against those obtained using BioNetGen (data not shown). BNGL-encoded specifications of these models, which can be processed by both DYNSTOC and BioNetGen, are available at the DYNSTOC web site. Below, we further validate DYNSTOC by considering two challenging test-case models that cannot be simulated using BioNetGen (except in special cases) but can be simulated using DYNSTOC and independent problem-specific approaches. These test cases demonstrate the ability of DYNSTOC to simulate multisite phosphorylation and multivalent binding dynamics.

3.1 Test case I: multisite phosphorylation

We consider an idealized rule-based model for a system in which autophosphorylation of a receptor tyrosine kinase (RTK) can generate a multitude of receptor phosphoforms and phosphorylation-dependent adapter-bound receptor states. The model captures the interactions of a cytosolic adapter protein with a dimer of RTKs, which are tightly associated. The adapter protein is comprised of a Src homology 2 (SH2) domain, and each receptor in a dimer is comprised of an active catalytic subunit and n autophosphorylation sites. When a site is phosphorylated, it can bind the SH2 domain of an adapter protein. The rules of the model are given in pseudo BNGL as follows:

$$\mathrm{D}(Y_{ij} \sim \mathrm{U}) \text{ -> } \mathrm{D}(Y_{ij} \sim \mathrm{P}) \tag{6a}$$

$$\mathrm{D}(Y_{ij} \sim \mathrm{P}) \text{ -> } \mathrm{D}(Y_{ij} \sim \mathrm{U}) \tag{6b}$$

$$\mathrm{D}(Y_{ij} \sim \mathrm{P}) + \mathrm{A(SH2)} \text{ <-> } \mathrm{D}(Y_{ij} \sim \mathrm{P!1}).\mathrm{A(SH2!1)} \tag{6c}$$

where 'D' denotes a receptor dimer, 'Y_{ij}' ($i = 1,\ldots,n$; $j = 1,2$) denotes the i-th tyrosine of the j-th receptor in a receptor dimer, 'U' denotes an unphosphorylated tyrosine, 'P' denotes a phosphorylated tyrosine, 'A' denotes an adapter protein and 'SH2' denotes the SH2 domain of an adapter protein. As usual in BNGL, the internal state label of a molecular component is prefixed by a tilde and a bond name is prefixed by an exclamation mark. Sharing of a bond name indicates that two molecular components are connected. The plus sign on the left-hand side of Equation (6c) indicates that the molecularity of reactions defined by this rule is two. The absence of a plus sign on the right-hand side indicates that reverse reactions have a molecularity of one. Reactions defined by the rules of Equations (6a) and (6b) also have a molecularity of one. The period on the right-hand side of Equation (6c) indicates joint membership in a complex. The above rules represent autophosphorylation of receptor tyrosines [Equation (6a)], dephosphorylation of receptor tyrosines via phosphatases not explicitly included in the model [Equation (6b)], and reversible adapter–receptor binding via SH2 domain recognition of phosphotyrosine [Equation (6c)]. An illustration of the model of Equations (6a–c) is available at the DYNSTOC web site.

The total number of rules defined in Equations (6a–c) is $6n$, and as can easily be confirmed, the number of chemical species implied by a rule set, N, is given by

$$N = 1 + 3^n + \binom{3^n}{2} = 1 + \frac{3^n(1+3^n)}{2} \tag{7}$$

In this count, one species corresponds to free adapter protein, A(SH2); 3^n species correspond to symmetric complexes and the rest correspond to asymmetric complexes. Note that a receptor has 3^n possible states because each of its n tyrosines has three possible states: free and unphosphorylated or phosphorylated and phosphorylated and bound. Thus, the rules of Equations (6a–c) tend to imply a large reaction network. However, because each receptor tyrosine is independent, this network can be characterized by a number of coupled ODEs derived from the law of mass action that is much less than N (Borisov *et al*., 2005; Conzelmann *et al*., 2006). As we will see, DYNSTOC is able to simulate Equations (6a–c) without taking advantage of this simplifying insight, and we can compare the results against those obtained from the reduced system of ODEs, which is given below.

For $i = 1, \ldots, n$ and $j = 1, 2$, we can write the following mass-action equations:

$$\frac{\mathrm{d}[U_{ij}]}{\mathrm{d}t} = -\phi_i[U_{ij}] + \delta_i[P_{ij}] \tag{8a}$$

$$\frac{\mathrm{d}[P_{ij}]}{\mathrm{d}t} = \phi_i[U_{ij}] - \delta_i[P_{ij}] - k_{+i}[P_{ij}] + k_{-i}[AP_{ij}] \tag{8b}$$

$$\frac{\mathrm{d}[AP_{ij}]}{\mathrm{d}t} = k_{+i}[P_{ij}][A] - k_{-i}[AP_{ij}] \tag{8c}$$

where $[U_{ij}]$ is the concentration of the i-th tyrosine in the j-th receptor in unphosphorylated form, $[P_{ij}]$ is the concentration of the i-th tyrosine in the j-th receptor in phosphorylated form and unbound to adapter, $[AP_{ij}]$ is the concentration of adapter protein bound to the i-th tyrosine in the j-th receptor, $[A]$ is the concentration of free adapter protein, ϕ_i is the apparent first-order rate constant for autophosphorylation of the i-th tyrosine in a receptor (we assume that autophosphorylation is substrate limited), δ_i is the apparent first-order rate constant for dephosphorylation of the i-th tyrosine in a receptor (we assume that phosphatases are present in excess), k_{+i} is the rate constant for binding of the adapter protein to the i-th tyrosine in a receptor, and k_{-i} is the rate constant for dissociation of the adapter protein from the i-th tyrosine in a receptor. If we assume that mass is conserved on the time scale of interest, we can also write the following equation:

$$[A] = [A_T] - \sum_{j=1}^{n} [AP_{ij}] \tag{9}$$

where $[A_T]$ is total concentration of adapter protein. Equations (8a–c) and (9) can be solved using standard numerical methods to obtain results that can be compared against DYNSTOC simulation results.

3.2 Test case II: multivalent binding

We consider an idealized rule-based model for a system in which multivalent ligand–receptor binding can generate a multitude of ligand-induced receptor aggregates. The model captures the interactions of a soluble trivalent ligand with a mobile cell-surface bivalent receptor (Yang *et al*., 2008). The rules of this model are given in BNGL as follows:

```
L(r,r,r) + R(1) -> L(r!1,r,r).R(1!1)          (10a)
L(r!+,r) + R(1) -> L(r!+,r!1).R(1!1)          (10b)
L(r!1).R(1!1) -> L(r) + R(1)                  (10c)
```

where 'L' denotes a ligand, 'R' denotes a receptor, 'r' denotes one of three identical binding sites on a ligand and 'l' denotes one of two identical binding sites on a receptor. The above rules represent capture of free ligand [Equation (10a)], receptor cross-linking [Equation (10b)] and ligand–receptor dissociation [Equation (10c)]. The model is relevant for studying several experimental systems (Bilgiçer *et al*., 2007; Posner *et al*., 2002, 2007; Sil *et al*., 2007). An illustration of the model of Equations (10a–c) is available at the DYNSTOC web site.

A large number of acyclic receptor aggregates can arise through the interactions represented by the rules of Equations (10a–c), and as a result, this model, depending on parameter values, can be difficult or impossible to simulate with conventional approaches (Hlavacek *et al*., 2006; Yang *et al*., 2008). Moreover, correct simulation of Equations (10a–c) requires enforcement of the '+' constraint of Equation (10b), which prohibits the formation of cyclic aggregates (Yang *et al*., 2008). This type of constraint on molecularity is difficult and sometimes expensive to enforce (Yang *et al*., 2008), but the need to enforce such a constraint is common, especially for rules that characterize aggregation phenomena.

Here, we will demonstrate how DYNSTOC simulations of Equations (10a–c) can be used to study the system of Posner *et al*. (2002). In particular, we will attempt to make a connection between cell-surface binding events and the cellular response to these events. The components of the system of Posner *et al*. (2002) include RBL cells, which express FcεRI (the high-affinity cell-surface receptor for IgE antibody), a model antigen containing three symmetrically arrayed 2,4-dinitrophenol (DNP) hapten groups, and a bivalent monoclonal anti-DNP IgE antibody. The trivalent antigen interacts with anti-DNP IgE–FcεRI complexes on RBL cells, which are long lived, to stimulate a robust cellular secretory response (Posner *et al*., 2002). Aggregation of FcεRI is known to trigger signaling events that can lead to degranulation (Metzger, 1992). We will assume, as in earlier work (Dembo and Goldstein, 1978), that the cellular secretory response to ligand correlates with the number of receptors in ligand-induced receptor aggregates at steady state. However, we will assume that receptor dimers are non-stimulatory because the bivalent analog of the trivalent ligand of Posner *et al*. (2002) does not elicit a secretory response. As shown in Figure 1, we can find values for parameters in the model of Yang *et al*. (2008), which we will call the TLBR model, such that ligand-induced receptor aggregation correlates with the secretory response to ligand. Below, we will use these parameter values to further investigate cell-surface ligand–receptor interactions.

3.3 Validation of simulation results

The results of simulations obtained using DYNSTOC and independent methods can be compared in Figure 2. The simulation results of Figure 2A were obtained by using DYNSTOC to process the rules of Equations (6a–c) and by numerical integration of

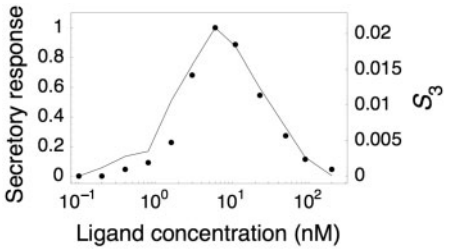

Fig. 1. The steady-state fraction of receptors in ligand-induced receptor aggregates containing three or more receptors, S_3, correlates with the secretory response of RBL cells at different doses of the trivalent ligand of Posner *et al*. (2002). Points represent measurements of secretion reported in Figure 4 of Posner *et al*. (2002). The line is obtained by using DYNSTOC to simulate the TLBR model [Equations (10a–c)] with the parameter values indicated below. There are six parameters in this model: ligand dose; N_R, the number of receptors per cell; V, the volume of extracellular fluid surrounding a single cell; k_{+1}, the single-site rate constant for ligand–receptor binding when ligand is freely diffusing; k_{+2}, the single-site rate constant for ligand–receptor binding when ligand is tethered to the cell surface; and k_{off}, the single-site rate constant for ligand–receptor dissociation. We set $V = 10^{-9}$ L, which corresponds to a cell density of 10^6 cells/ml; we set $k_{\text{off}} = 0.01\ \text{s}^{-1}$, which is consistent with assays of binding of monovalent hapten to anti-DNP IgE (Erickson *et al*., 1987; Xu *et al*., 1998); and we set $N_R = 300\,000$/cell, which is consistent with assays of the number of FcεRI on RBL cells (Erickson *et al*., 1987). We systematically varied k_{+1} and k_{+2} in a grid search to find values for which S_3 is maximal at the same ligand dose that yields maximal secretory response. The line shown in this figure is calculated using $k_{+1} = 3.6 \times 10^5\ \text{M}^{-1}\ \text{s}^{-1}$ and $k_{+2}N_R = 9 \times 10^{-4}\ \text{s}^{-1}$. To speed calculations, we considered only 1% of the volume surrounding a cell in simulations.

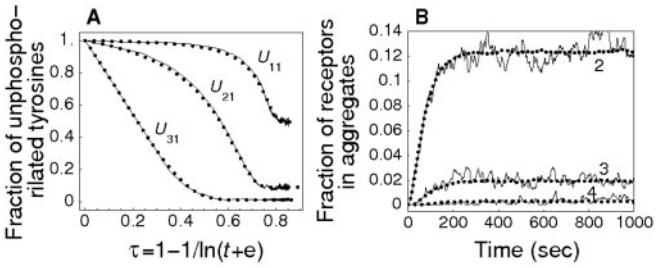

Fig. 2. Validation of DYNSTOC simulation results. (**A**) Time courses of tyrosine phosphorylation calculated by using numerical integration to solve Equations (8a–c) and (9) for $n = 6$ (dotted lines) and by using DYNSTOC to simulate the corresponding set of rules given by Equations (6a–c) (solid lines). Note the transformation of time t (s). The initial data are $[A] = [A_T] = 0.12\ \mu$M and $[U_{ij}] = 1.2\ \mu$M, $[P_{ij}] = 0$, and $[AP_{ij}] = 0$ for $i = 1, \ldots, 6$ and $j = 1, 2$. Additional parameters have the following values: $V = 1.4 \times 10^{-12}$ L (cell volume); $k_{+1} = k_{+2} = 7.6 \times 10^6\ \text{M}^{-1}\text{s}^{-1}$, $k_{+3} = k_{+4} = 1.7 \times 10^6\ \text{M}^{-1}\text{s}^{-1}$, $k_{+5} = k_{+6} = 6.7 \times 10^6\ \text{M}^{-1}\text{s}^{-1}$; $k_{-i} = 0.3\ \text{s}^{-1}$ for $i = 1, \ldots, 6$; $\phi_1 = \phi_2 = 0.01s^{-1}$, $\phi_3 = \phi_4 = 0.1s^{-1}$, $\phi_5 = \phi_6 = 0.8s^{-1}$; and $\delta_I = 0.01s^{-1}$ for $i = 1, \ldots, 6$. (**B**) Time courses of ligand-induced receptor aggregation after the addition of ligand (6 nM) according to Equations (10a–c). Time courses are calculated by using a problem-specific implementation of the YMFH method (dotted lines) and DYNSTOC (solid lines). Parameter values are the same as for Figure 1. Time courses are shown for aggregates containing 2, 3 and 4 receptors. To speed calculations, we considered only 1% of the relevant volume in DYNSTOC simulations; we considered a larger volume in simulations using the YMFH method to reduce noise. The model/simulation-specification files processed by DYNSTOC to produce the results shown here (testcase1.bngl and testcase2.bngl) are available at the DYNSTOC web site.

Equations (8a–c) and (9) for $n = 6$ [$N = 266\,086$; Equation (7)]. Interestingly, the time courses of Figure 2A indicate that ordered phosphorylation of tyrosines is possible for purely kinetic reasons. The simulation results of Figure 2B were obtained by using DYNSTOC to process the rules of Equations (10a–c) and by using a problem-specific implementation of the (general) method of Yang *et al*. (2008), which we will call the YMFH method, to process the same rule set. As can be seen, DYNSTOC produces results that are consistent with those obtained independently. Interestingly, the time courses of Figure 2B suggest that receptor trimers are predominantly responsible for the RBL secretory response to the trivalent ligand of Posner *et al*. (2002). The results of Figure 2 serve not only to validate DYNSTOC, but also to demonstrate that DYNSTOC is capable of simulating large-scale rule-based models.

3.4 Efficiency

Figure 3 illustrates the efficiency of DYNSTOC relative to the YMFH method. Both methods were used to simulate the rules of Equations (10a–c) over ranges of model parameter values that affect the complexity of simulation. Increasing the number of receptors N_R (Fig. 3A) increases the frequency of reactions, and increasing the dimensionless parameter $\beta = N_R k_{+2}/k_{\text{off}}$ (Fig. 3B) increases the average size of ligand-induced receptor aggregates at equilibrium (Goldstein and Perelson, 1984). As can be seen, the computational cost of simulating Equations (10a–c) with

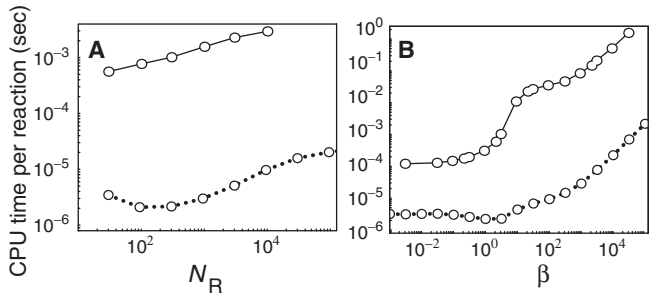

Fig. 3. Efficiency of DYNSTOC. We compare DYNSTOC (solid lines) and a problem-specific implementation of the YMFH method (dotted lines); these methods are used to simulate Equations (10a–c). (**A**) Scaling of computational cost with system size, where size is measured by N_R, the number of cell-surface receptors. Additional parameter values: $V = 10^{-12}$ L; $k_{+1} = 1.8 \times 10^5\ \text{M}^{-1}\text{s}^{-1}$; $k_{+2}N_R^{\text{ref}} = 0.03\ \text{s}^{-1}$, where $N_R^{ref} = 300$;$k_{\text{off}} = 0.01\ \text{s}^{-1}$; and $N_L = 4200$. (**B**) Scaling of computational cost with dimensionless parameter $\beta = N_R k_{+2}/k_{\text{off}}$, which controls the (equilibrium) extent of ligand-induced receptor aggregation. The value of β was adjusted by varying k_{+2} while holding $N_R = 300$, and $k_{\text{off}} = 0.01\ \text{s}^{-1}$ fixed. Additional parameter values: $V = 10^{-12}$ L, $N_L = 4200$ and $k_{+1} = 1.8 \times 10^7\ \text{M}^{-1}\text{s}^{-1}$. In each panel, the y-axis indicates the total CPU time per reaction event required to simulate the kinetics of the TLBR model from time $t = 0$ s to 1000 s with all ligand initially free. Parameter values used in simulations are the same as those used by Yang *et al*. (2008). See the file tlbr.bngl at the DYNSTOC web site.

DYNSTOC scales as a function of simulation complexity similarly to the YMFH method. However, the cost of DYNSTOC simulations can be orders of magnitude greater. The results of Figure 3 are not unexpected, because time steps in the method of DYNSTOC are fixed and must generally be smaller than in the YMFH method, which involves sampling of time steps from an exponential distribution, as in the method of Gillespie (2007). It should be emphasized that DYNSTOC is a general-purpose simulation tool, whereas the implementation of the YMFH method being considered here is problem specific. It should also be remembered that simulation of the rules of Equations (10a–c) is impractical with conventional methods (Hlavacek *et al*., 2006; Yang *et al*., 2008).

4 CLOSING REMARKS

Languages, such as BNGL, provide a means for the specification of kinetic models for signal-transduction and other biochemical systems in terms of rules for molecular interactions. However, because of combinatorial complexity (Hlavacek *et al*., 2003, 2006), the biochemical reaction networks implied by typical rule sets are large scale. The process of deriving a network from rules is expensive, and rule-derived networks (if they can be practically generated from rules) are difficult to simulate using conventional ODE-based and stochastic simulation approaches. To address this problem, we have generalized the agent-based simulation method of StochSim and implemented the generalized method in software called DYNSTOC, which can interpret BNGL-encoded model specifications.

The generalized simulation method presented here differs from the original StochSim method in a number of ways. For example, model specification and reporting of simulation results are greatly eased by the ability to use BNGL. However, the main advance is reformulation of the StochSim method to enable explicit tracking of the connectivity of molecules in molecular complexes. This advance is enabled by the use of graphs to represent molecules and molecular complexes. The generalized method is also capable of accounting for a richer variety of reaction types, such as intramolecular association reactions, which, with the exception of special cases, cannot be considered within the original StochSim framework.

The results of Figure 3 suggest that the StochSim/DYNSTOC simulation method is less efficient than the YMFH method. These results provide motivation for development of general-purpose software implementing the YMFH method, and DYNSTOC should be useful for validating such software when available. Although other simulation methods may be more efficient, DYNSTOC should still be useful for simulating a wide variety of biochemical systems, and we find the capabilities of DYNSTOC demonstrated here to be quite exciting.

ACKNOWLEDGEMENTS

We thank Guy Yosiphon for helpful discussions.

Funding: National Institutes of Health (AI35997, CA109552, GM076570 and RR18754); Department of Energy (DOE) contract DE-AC52-06NA25396; Arizona Biomedical Research Commission.

Conflict of Interest: none declared.

REFERENCES

Andrei,O. and Kirchner,H. (2008) Graph rewriting strategies for modeling biochemical networks. In Negru,V. *et al*. (eds) *Proceedings of the Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*. IEEE, Piscataway, NJ, pp. 407–414.

Bilgiçer,B. *et al*. (2007) A synthetic trivalent hapten that aggregates anti-2,4-DNP IgG into bicyclic trimers. *J. Am. Chem. Soc.*, **129**, 3722–3728.

Blinov,M.L. *et al*. (2004) BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains. *Bioinformatics*, **20**, 3289–3291.

Blinov,M.L. *et al*. (2006) Graph theory for rule-based modeling of biochemical networks. *Lect. Notes Comput. Sci.*, **4230**, 89–106.

Borisov,N.M. *et al*. (2005) Signaling through receptors and scaffolds: independent interactions reduce combinatorial complexity. *Biophys. J.*, **89**, 951–966.

Chatterjee,A. and Vlachos,D.G. (2007) An overview of spatial microscopic and accelerated kinetic Monte Carlo methods. *J. Comput. Aided Mater. Des.*, **14**, 253–308.

Conzelmann,H. *et al*. (2006) A domain-oriented approach to the reduction of combinatorial complexity in signal transduction networks. *BMC Bioinformatics*, **7**, 34.

Danos,V. (2007) Agile modelling of cellular signalling. *AIP Conf. Proc.*, **963**, 611–614.

Danos,V. and Laneve,C. (2004) Formal molecular biology. *Theor. Comput. Sci.*, **325**, 69–110.

Danos,V. *et al*. (2007a) Rule-based modelling of cellular signalling. *Lect. Notes Comput. Sci.*, **4703**, 17–41.

Danos,V. *et al*. (2007b) Scalable simulation of cellular signaling networks. *Lect. Notes Comput. Sci.*, **4807**, 139–157.

Dembo,M. and Goldstein,B. (1978) Theory of equilibrium binding of symmetric bivalent haptens to cell surface antibody: application to histamine release from basophils. *J. Immunol.*, **121**, 345–353.

Dematté,L. *et al*. (2008) The BlenX language: a tutorial. *Lect. Notes Comput. Sci.*, **5016**, 313–365.

Erickson,J. *et al*. (1987) The effect of receptor density on the forward rate constant for binding of ligands to cell surface receptors. *Biophys. J.*, **52**, 657–662.

Faeder,J.R. *et al*. (2005a) Rule-based modeling of biochemical networks. *Complexity*, **10**, 22–41.

Faeder,J.R. *et al*. (2005b) Graphical rule-based representation of signal-transduction networks. In Liebrock,L.M. (ed.) *Proceeings of the 2005 ACM Symposium on Applied Computing*. ACM Press, New York, pp. 133–140.

Faeder,J.R. *et al*. (in press) Rule-based modeling of biochemical systems with BioNetGen. *Methods Mol. Biol.*

Gillespie,D.T. (2007) Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, **58**, 35–55.

Goldstein,B. and Perelson,A.S. (1984) Equilibrium theory for the clustering of bivalent cell surface receptors by trivalent ligands. Application to histamine release from basophils. *Biophys. J.*, **45**, 1109–1123.

Hlavacek,W.S. *et al*. (2003) The complexity of complexes in signal transduction. *Biotechnol. Bioeng*., **84**, 783–794.

Hlavacek,W.S. *et al*. (2006) Rules for modeling signal-transduction systems. *Sci. STKE*, **2006**, re6.

Kitano,H. *et al*. (2005) Using process diagrams for the graphical representation of biological networks. *Nat. Biotechnol.*, **23**, 961–966.

Le Novère,N. and Shimizu,T.S. (2001) StochSim: modelling of stochastic biomolecular processes. *Bioinformatics*, **17**, 575–576.

Li,H. *et al*. (2008) Algorithms and software for stochastic simulation of biochemical reacting systems. *Biotechnol. Prog.*, **24**, 56–61.

Lok,L. and Brent,R. (2005) Automatic generation of cellular reaction networks with Moleculizer 1.0. *Nat. Biotechnol.*, **23**, 131–136.

Metzger,H. (1992) Transmembrane signaling: the joy of aggregation. *J. Immunol.*, **149**, 1477–1487.

Mjolsness,E. and Yosiphon,G. (2006) Stochastic process semantics for dynamical grammars. *Ann. Math. Artif. Intell.*, **47**, 329–395.

Morton-Firth,C.J. and Bray,D. (1998) Predicting temporal fluctuations in an intracellular signalling pathway. *J. Theor. Biol.*, **192**, 117–128.

Mu,F. *et al*. (2007) Carbon-fate maps for metabolic reactions. *Bioinformatics*, **23**, 3193–3199.

Posner,R.G. *et al*. (2002) A quantitative approach for studying IgE-FcεRI aggregation. *Mol. Immunol.*, **38**, 1221–1228.

Posner,R.G. *et al*. (2007) Trivalent antigens for degranulation of mast cells. *Org. Lett.*, **9**, 3551–3554.

Schulze,T.P. (2007) Efficient kinetic Monte Carlo simulation. *J. Comput. Phys.*, **227**, 2455–2462.

Shimizu,T.S. and Bray,D. (2001) Computational cell biology—the stochastic approach. In Kitano,H. (ed.) *Foundations of Systems Biology*. Ch. 10, MIT Press, Cambridge, MA.

Sil,D. *et al*. (2007) Trivalent ligands with rigid DNA spacers reveal structural requirements for IgE receptor signaling in RBL mast cells. *ACS Chem. Biol.*, **2**, 674–684.

Slepoy,A. *et al*. (2008) A constant-time kinetic Monte Carlo algorithm for simulation of large biochemical reaction networks. *J. Chem. Phys.*, **128**, 205101.

Xu,K. *et al*. (1998) Kinetics of multivalent antigen DNP-BSA binding to IgE-FcεRI in relationship to the stimulated tyrosine phosphorylation of FcεRI. *J. Immunol.*, **160**, 3225–3235.

Yang,J. *et al*. (2008) Kinetic Monte Carlo method for rule-based modeling of biochemical networks. *Phys. Rev. E*, **78**, 031910.